# Self-tuning extremum control

Kit Tiyapan

The term *extremum control* (Wellstead and Zarrop 1991) is a synonym for optimization. It means making a process operate at the optimal point at all times. The optimal point is normally obtained by extremizing a performance index. Self-tuning extremum control is useful when dealing with systems with performance index measured in a noisy environment.

*Hill-climbing* is a very useful technique. We can imagine a performance function as being a hill.

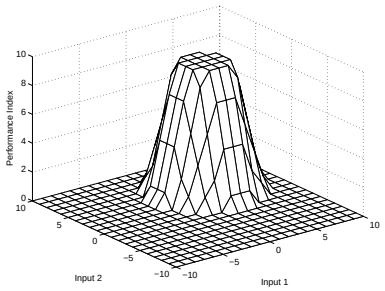


Figure 1: *Example of a relationship between a performance index and two inputs*

When we are at a specific point (control value) we can calculate the performance index at that point. The performance index we obtain represents the height of the hill there. If we compare this performance index with those of the points in our close neighbourhood we will know which direction to move to in order to increase our performance index. The process is carried out recursively until we reach the top of the hill where no neighbouring point leads to any improvement in performance.

Depending on how much knowledge of the hill we have at the start of our quest, we may be able to fix some of the parameters without any adverse effect on the usefulness of the result obtained (Tiyapan 1995). For example, if our model is a quadratic one, we may be able to fix the coefficient of the square term. In so doing, our estimated hill will still give the correct information of the position of the hill top. This is enough for most practical situations where we only need to know the position (the value of the input to system) of the hill top (the point of maximum performance) without caring too much about the height of the hill (the value of the actual performance index).

*Self-tuning control* (Wellstead and Zarrop 1991) is closely related to adaptive control. Here the input and output of the system are measured and used to adjust model parameters. Algorithms used in self-tuning controllers rely heavily upon the theory of recursive least-square. With the recursive least squares algorithm it is possible to extend the approach to multi-input case (Zarrop and Rommens 1993) and hence to an object-location problem (Tiyapan 1995).

When there exists some reference model describing the performance, the output from the reference model, together with measured input/output data can be used to monitor system parameters. This is called a *model-based* algorithm (Wellstead and Zarrop 1991). In the neighbourhood of extremum a quadratic model is a good representation of the hill top. If the hill is smooth at the summit, this model represents the first few terms of a Taylor's series for the

performance function (Tiyapan 1995). Because the recursive least-square algorithm use the difference in extimated value compared with the actual value, the smoothness of the hill-top implies the slowness of convergence of the algorithm when approaching the summit. We say that the algorithm is going to sleep. To prevent it from doing so, we add a small signal, usually with zero-mean, called a *dither signal* to our input.

From simulation (Tiyapan 1995), we can see that the speed of convergence is improved when we use a dither signal with a larger variance. The symmetry of the hill is also important. If a hill is asymmetric, we need smaller dither signal in order to be able to have better estimate. Therefore, when the performance function is asymmetric, we have to do some trade-off between the speed of convergence and the accuracy of estimated value.

## References


Tiyapan, K. (1995). Self-tuning extremum control. Design exercise report. Control System Centre, University of Manchester Institute of Science and Technology. Manchester, U.K.

Wellstead, P. E. and M. B. Zarrop (1991). *Self-tuning systems*. John Wiley & Sons Ltd.

Zarrop, M. B. and M. J. J. J. Rommens (1993). 'Convergence of a multi-input adaptive extremum controller'. *IEE Proceedings-D* **140**(2), 65–69.